****A Hybrid LSTM-PPO Framework for High-Accuracy Options Trading Based on Technical Setups in the Indian Market****

Build a smart AI system that takes any technical strategy (like gap-up, trap candle, impulsive), and uses LSTM to classify the signal's strength and PPO/DQN to decide whether to trade CE, PE, or skip — all based on PnL reward and market context.

Install Required Dependancies

In [16]:
```
!pip3 install smartapi-python --upgrade stable-baselines3
!pip3 install pandas_ta


!pip3 install stable-baselines3
!pip3 install shimmy
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: smartapi-python in /Users/nil/Library/Python/3.
9/lib/python/site-packages (1.5.5)
Requirement already satisfied: stable-baselines3 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (2.6.0)
Requirement already satisfied: requests>=2.18.4 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from smartapi-python) (2.32.3)
Requirement already satisfied: six>=1.11.0 in /Library/Developer/CommandLineTo
ols/Library/Frameworks/Python3.framework/Versions/3.9/lib/python3.9/site-packa
ges (from smartapi-python) (1.15.0)
Requirement already satisfied: python-dateutil>=2.6.1 in /Users/nil/Library/Py
thon/3.9/lib/python/site-packages (from smartapi-python) (2.9.0.post0)
Requirement already satisfied: gymnasium<1.2.0,>=0.29.1 in /Users/nil/Library/
Python/3.9/lib/python/site-packages (from stable-baselines3) (1.1.1)
Requirement already satisfied: numpy<3.0,>=1.20 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from stable-baselines3) (1.26.4)
Requirement already satisfied: torch<3.0,>=2.3 in /Users/nil/Library/Python/3.
9/lib/python/site-packages (from stable-baselines3) (2.6.0)
Requirement already satisfied: cloudpickle in /Users/nil/Library/Python/3.9/li
b/python/site-packages (from stable-baselines3) (3.1.0)
Requirement already satisfied: pandas in /Users/nil/Library/Python/3.9/lib/pyt
hon/site-packages (from stable-baselines3) (2.2.3)
Requirement already satisfied: matplotlib in /Users/nil/Library/Python/3.9/li
b/python/site-packages (from stable-baselines3) (3.9.3)
Requirement already satisfied: importlib-metadata>=4.8.0 in /Users/nil/Librar
y/Python/3.9/lib/python/site-packages (from gymnasium<1.2.0,>=0.29.1->stable-b
aselines3) (8.5.0)
Requirement already satisfied: typing-extensions>=4.3.0 in /Users/nil/Library/
Python/3.9/lib/python/site-packages (from gymnasium<1.2.0,>=0.29.1->stable-bas
elines3) (4.12.2)
Requirement already satisfied: farama-notifications>=0.0.1 in /Users/nil/Libra
ry/Python/3.9/lib/python/site-packages (from gymnasium<1.2.0,>=0.29.1->stable-
baselines3) (0.0.4)
Requirement already satisfied: charset-normalizer<4,>=2 in /Users/nil/Library/
Python/3.9/lib/python/site-packages (from requests>=2.18.4->smartapi-python)
(3.4.0)
Requirement already satisfied: idna<4,>=2.5 in /Users/nil/Library/Python/3.9/l
ib/python/site-packages (from requests>=2.18.4->smartapi-python) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /Users/nil/Library/Pytho
n/3.9/lib/python/site-packages (from requests>=2.18.4->smartapi-python) (2.2.
3)
Requirement already satisfied: certifi>=2017.4.17 in /Users/nil/Library/Pytho
n/3.9/lib/python/site-packages (from requests>=2.18.4->smartapi-python) (2024.
8.30)
Requirement already satisfied: filelock in /Users/nil/Library/Python/3.9/lib/p
ython/site-packages (from torch<3.0,>=2.3->stable-baselines3) (3.18.0)
Requirement already satisfied: networkx in /Users/nil/Library/Python/3.9/lib/p
ython/site-packages (from torch<3.0,>=2.3->stable-baselines3) (3.2.1)
Requirement already satisfied: jinja2 in /Users/nil/Library/Python/3.9/lib/pyt
hon/site-packages (from torch<3.0,>=2.3->stable-baselines3) (3.1.4)
Requirement already satisfied: fsspec in /Users/nil/Library/Python/3.9/lib/pyt
hon/site-packages (from torch<3.0,>=2.3->stable-baselines3) (2025.3.2)
Requirement already satisfied: sympy==1.13.1 in /Users/nil/Library/Python/3.9/
lib/python/site-packages (from torch<3.0,>=2.3->stable-baselines3) (1.13.1)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /Users/nil/Library/Pytho
n/3.9/lib/python/site-packages (from sympy==1.13.1->torch<3.0,>=2.3->stable-ba
```

```
selines3) (1.3.0)
Requirement already satisfied: contourpy>=1.0.1 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (1.3.0)
Requirement already satisfied: cycler>=0.10 in /Users/nil/Library/Python/3.9/l
ib/python/site-packages (from matplotlib->stable-baselines3) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (4.55.0)
Requirement already satisfied: kiwisolver>=1.3.1 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (1.4.7)
Requirement already satisfied: packaging>=20.0 in /Users/nil/Library/Python/3.
9/lib/python/site-packages (from matplotlib->stable-baselines3) (24.2)
Requirement already satisfied: pillow>=8 in /Users/nil/Library/Python/3.9/lib/
python/site-packages (from matplotlib->stable-baselines3) (11.0.0)
Requirement already satisfied: pyparsing>=2.3.1 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (3.2.0)
Requirement already satisfied: importlib-resources>=3.2.0 in /Users/nil/Librar
y/Python/3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (6.
4.5)
Requirement already satisfied: pytz>=2020.1 in /Users/nil/Library/Python/3.9/l
ib/python/site-packages (from pandas->stable-baselines3) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /Users/nil/Library/Python/3.
9/lib/python/site-packages (from pandas->stable-baselines3) (2024.2)
Requirement already satisfied: zipp>=3.20 in /Users/nil/Library/Python/3.9/li
b/python/site-packages (from importlib-metadata>=4.8.0->gymnasium<1.2.0,>=0.2
9.1->stable-baselines3) (3.21.0)
Requirement already satisfied: MarkupSafe>=2.0 in /Users/nil/Library/Python/3.
9/lib/python/site-packages (from jinja2->torch<3.0,>=2.3->stable-baselines3)
(3.0.2)

[notice] A new release of pip is available: 24.3.1 -> 25.0.1
[notice] To update, run: /Library/Developer/CommandLineTools/usr/bin/python3 -
m pip install --upgrade pip
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pandas_ta in /Users/nil/Library/Python/3.9/lib/
python/site-packages (0.3.14b0)
Requirement already satisfied: pandas in /Users/nil/Library/Python/3.9/lib/pyt
hon/site-packages (from pandas_ta) (2.2.3)
Requirement already satisfied: numpy>=1.22.4 in /Users/nil/Library/Python/3.9/
lib/python/site-packages (from pandas->pandas_ta) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in /Users/nil/Library/Py
thon/3.9/lib/python/site-packages (from pandas->pandas_ta) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /Users/nil/Library/Python/3.9/l
ib/python/site-packages (from pandas->pandas_ta) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /Users/nil/Library/Python/3.
9/lib/python/site-packages (from pandas->pandas_ta) (2024.2)
Requirement already satisfied: six>=1.5 in /Library/Developer/CommandLineTool
s/Library/Frameworks/Python3.framework/Versions/3.9/lib/python3.9/site-package
s (from python-dateutil>=2.8.2->pandas->pandas_ta) (1.15.0)

[notice] A new release of pip is available: 24.3.1 -> 25.0.1
[notice] To update, run: /Library/Developer/CommandLineTools/usr/bin/python3 -
m pip install --upgrade pip
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: stable-baselines3 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (2.6.0)
Requirement already satisfied: gymnasium<1.2.0,>=0.29.1 in /Users/nil/Library/
```

```
Python/3.9/lib/python/site-packages (from stable-baselines3) (1.1.1)
Requirement already satisfied: numpy<3.0,>=1.20 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from stable-baselines3) (1.26.4)
Requirement already satisfied: torch<3.0,>=2.3 in /Users/nil/Library/Python/3.
9/lib/python/site-packages (from stable-baselines3) (2.6.0)
Requirement already satisfied: cloudpickle in /Users/nil/Library/Python/3.9/li
b/python/site-packages (from stable-baselines3) (3.1.0)
Requirement already satisfied: pandas in /Users/nil/Library/Python/3.9/lib/pyt
hon/site-packages (from stable-baselines3) (2.2.3)
Requirement already satisfied: matplotlib in /Users/nil/Library/Python/3.9/li
b/python/site-packages (from stable-baselines3) (3.9.3)
Requirement already satisfied: importlib-metadata>=4.8.0 in /Users/nil/Librar
y/Python/3.9/lib/python/site-packages (from gymnasium<1.2.0,>=0.29.1->stable-b
aselines3) (8.5.0)
Requirement already satisfied: typing-extensions>=4.3.0 in /Users/nil/Library/
Python/3.9/lib/python/site-packages (from gymnasium<1.2.0,>=0.29.1->stable-bas
elines3) (4.12.2)
Requirement already satisfied: farama-notifications>=0.0.1 in /Users/nil/Libra
ry/Python/3.9/lib/python/site-packages (from gymnasium<1.2.0,>=0.29.1->stable-
baselines3) (0.0.4)
Requirement already satisfied: filelock in /Users/nil/Library/Python/3.9/lib/p
ython/site-packages (from torch<3.0,>=2.3->stable-baselines3) (3.18.0)
Requirement already satisfied: networkx in /Users/nil/Library/Python/3.9/lib/p
ython/site-packages (from torch<3.0,>=2.3->stable-baselines3) (3.2.1)
Requirement already satisfied: jinja2 in /Users/nil/Library/Python/3.9/lib/pyt
hon/site-packages (from torch<3.0,>=2.3->stable-baselines3) (3.1.4)
Requirement already satisfied: fsspec in /Users/nil/Library/Python/3.9/lib/pyt
hon/site-packages (from torch<3.0,>=2.3->stable-baselines3) (2025.3.2)
Requirement already satisfied: sympy==1.13.1 in /Users/nil/Library/Python/3.9/
lib/python/site-packages (from torch<3.0,>=2.3->stable-baselines3) (1.13.1)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /Users/nil/Library/Pytho
n/3.9/lib/python/site-packages (from sympy==1.13.1->torch<3.0,>=2.3->stable-ba
selines3) (1.3.0)
Requirement already satisfied: contourpy>=1.0.1 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (1.3.0)
Requirement already satisfied: cycler>=0.10 in /Users/nil/Library/Python/3.9/l
ib/python/site-packages (from matplotlib->stable-baselines3) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (4.55.0)
Requirement already satisfied: kiwisolver>=1.3.1 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (1.4.7)
Requirement already satisfied: packaging>=20.0 in /Users/nil/Library/Python/3.
9/lib/python/site-packages (from matplotlib->stable-baselines3) (24.2)
Requirement already satisfied: pillow>=8 in /Users/nil/Library/Python/3.9/lib/
python/site-packages (from matplotlib->stable-baselines3) (11.0.0)
Requirement already satisfied: pyparsing>=2.3.1 in /Users/nil/Library/Python/
3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (3.2.0)
Requirement already satisfied: python-dateutil>=2.7 in /Users/nil/Library/Pyth
on/3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (2.9.0.po
st0)
Requirement already satisfied: importlib-resources>=3.2.0 in /Users/nil/Librar
y/Python/3.9/lib/python/site-packages (from matplotlib->stable-baselines3) (6.
4.5)
Requirement already satisfied: pytz>=2020.1 in /Users/nil/Library/Python/3.9/l
ib/python/site-packages (from pandas->stable-baselines3) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /Users/nil/Library/Python/3.
```

```
9/lib/python/site-packages (from pandas->stable-baselines3) (2024.2)
Requirement already satisfied: zipp>=3.20 in /Users/nil/Library/Python/3.9/li
b/python/site-packages (from importlib-metadata>=4.8.0->gymnasium<1.2.0,>=0.2
9.1->stable-baselines3) (3.21.0)
Requirement already satisfied: six>=1.5 in /Library/Developer/CommandLineTool
s/Library/Frameworks/Python3.framework/Versions/3.9/lib/python3.9/site-package
s (from python-dateutil>=2.7->matplotlib->stable-baselines3) (1.15.0)
Requirement already satisfied: MarkupSafe>=2.0 in /Users/nil/Library/Python/3.
9/lib/python/site-packages (from jinja2->torch<3.0,>=2.3->stable-baselines3)
(3.0.2)

[notice] A new release of pip is available: 24.3.1 -> 25.0.1
[notice] To update, run: /Library/Developer/CommandLineTools/usr/bin/python3 -
m pip install --upgrade pip
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: shimmy in /Users/nil/Library/Python/3.9/lib/pyt
hon/site-packages (2.0.0)
Requirement already satisfied: numpy>=1.18.0 in /Users/nil/Library/Python/3.9/
lib/python/site-packages (from shimmy) (1.26.4)
Requirement already satisfied: gymnasium>=1.0.0a1 in /Users/nil/Library/Pytho
n/3.9/lib/python/site-packages (from shimmy) (1.1.1)
Requirement already satisfied: cloudpickle>=1.2.0 in /Users/nil/Library/Pytho
n/3.9/lib/python/site-packages (from gymnasium>=1.0.0a1->shimmy) (3.1.0)
Requirement already satisfied: importlib-metadata>=4.8.0 in /Users/nil/Librar
y/Python/3.9/lib/python/site-packages (from gymnasium>=1.0.0a1->shimmy) (8.5.
0)
Requirement already satisfied: typing-extensions>=4.3.0 in /Users/nil/Library/
Python/3.9/lib/python/site-packages (from gymnasium>=1.0.0a1->shimmy) (4.12.2)
Requirement already satisfied: farama-notifications>=0.0.1 in /Users/nil/Libra
ry/Python/3.9/lib/python/site-packages (from gymnasium>=1.0.0a1->shimmy) (0.0.
4)
Requirement already satisfied: zipp>=3.20 in /Users/nil/Library/Python/3.9/li
b/python/site-packages (from importlib-metadata>=4.8.0->gymnasium>=1.0.0a1->sh
immy) (3.21.0)

[notice] A new release of pip is available: 24.3.1 -> 25.0.1
[notice] To update, run: /Library/Developer/CommandLineTools/usr/bin/python3 -
m pip install --upgrade pip
```

In [17]:
```
from SmartApi import SmartConnect
import pyotp
import http.client
import json
from datetime import datetime, timedelta
import pandas as pd
import socket
import requests
import uuid
from ta.trend import EMAIndicator, MACD, ADXIndicator
from ta.momentum import RSIIndicator, StochasticOscillator
from ta.volatility import BollingerBands
from sklearn.preprocessing import MinMaxScaler
```

In [18]:
```
def Login_To_Angleone():
    api_key = "WUCyfVgK" # Using the first API Key
    smartAPI = SmartConnect(api_key)
```

```
    try:

        totp = pyotp.TOTP("2AMWG2Z2FZ3ZOFBJATC7EFDWFY").now()
        data = smartAPI.generateSession("N274681", "6499", totp)
        if not data['status']:
            return None

        jwtToken = data['data']['jwtToken']
        refreshToken = data['data']['refreshToken']
        feedToken = data['data']['feedToken']
        return jwtToken, refreshToken, feedToken, smartAPI
    except Exception as e:
        print(f"Error: {e}")
        return None
```

In [ ]:
```
jwtToken, refreshToken, feedToken, smartAPI = Login_To_Angleone()
```

In [19]:
```
def Get_IP_Info():
    s = socket.socket(socket.AF_INET,  socket.SOCK_DGRAM)

    local_IP =  "127.0.0.1"
    Public_IP = requests.get("https://api.ipify.org").text

    try:
        s.connect(("10.254.254.254",1)) # We use 10.254.254.254 to trick syste
        local_IP = s.getsockname()[0]
    finally:
        s.close()
    return local_IP, Public_IP,':'.join(['{:02x}'.format((uuid.getnode() >> el
```

In [20]:
```
from time import sleep


def fetch_5_years_5min_candle(authToken, index):


    # Get stock token
    Stock_Token = "99926000" # For Nifty

    conn = http.client.HTTPSConnection("apiconnect.angelbroking.com")
    local_IP, public_IP, mac_address = Get_IP_Info()
    apikey = "WUCyfVgK"

    # Define time range
    end_date = datetime.now().replace(second=0, microsecond=0)
    start_date = end_date - timedelta(days=365 * 5)  # 5 years ago

    all_candles = []

    while start_date < end_date:
        batch_end = min(start_date + timedelta(days=30), end_date)

        payload = json.dumps({
            "exchange": "NSE",
            "symboltoken": Stock_Token,
```

```
            "interval": "FIVE_MINUTE",
            "fromdate": start_date.strftime("%Y-%m-%d %H:%M"),
            "todate": batch_end.strftime("%Y-%m-%d %H:%M")
        })

        headers = {
            'X-PrivateKey': apikey,
            'Authorization': authToken,
            'X-SourceID': 'WEB',
            'X-ClientLocalIP': local_IP,
            'X-ClientPublicIP': public_IP,
            'X-MACAddress': mac_address,
            'X-UserType': 'USER',
            'Accept': 'application/json',
            'Content-Type': 'application/json'
        }

        try:
            conn.request("POST", "/rest/secure/angelbroking/historical/v1/getC
            res = conn.getresponse()
            data = res.read().decode("utf-8")
            response_json = json.loads(data)
        except Exception as e:
            print(f"Request failed from {start_date} to {batch_end}: {e}")
            break

        if 'data' in response_json and response_json['data']:
            all_candles.extend(response_json['data'])
            print(f"✅ Fetched: {start_date.strftime('%Y-%m-%d')} to {batch_er
        else:
            print(f"⚠️ No data for {start_date.strftime('%Y-%m-%d')} to {batch

        # Move to next batch
        start_date = batch_end
        sleep(1.2)  # avoid rate limiting (optional, adjust as needed)

    conn.close()

    if not all_candles:
        return None

    # Convert to DataFrame
    df = pd.DataFrame(all_candles, columns=['timestamp', 'open', 'high', 'low'
    df['timestamp'] = pd.to_datetime(df['timestamp'])
    df[['open', 'high', 'low', 'close', 'volume']] = df[['open', 'high', 'low'

    return df
```

In [51]:
```
df = fetch_5_years_5min_candle(jwtToken, "NIFTY")
```

```
✅ Fetched: 2020-04-19 to 2020-05-19 - Candles: 1575
✅ Fetched: 2020-05-19 to 2020-06-18 - Candles: 1575
✅ Fetched: 2020-06-18 to 2020-07-18 - Candles: 1575
✅ Fetched: 2020-07-18 to 2020-08-17 - Candles: 1575
✅ Fetched: 2020-08-17 to 2020-09-16 - Candles: 1650
✅ Fetched: 2020-09-16 to 2020-10-16 - Candles: 1575
✅ Fetched: 2020-10-16 to 2020-11-15 - Candles: 1514
✅ Fetched: 2020-11-15 to 2020-12-15 - Candles: 1500
✅ Fetched: 2020-12-15 to 2021-01-14 - Candles: 1575
✅ Fetched: 2021-01-14 to 2021-02-13 - Candles: 1500
✅ Fetched: 2021-02-13 to 2021-03-15 - Candles: 1451
✅ Fetched: 2021-03-15 to 2021-04-14 - Candles: 1425
✅ Fetched: 2021-04-14 to 2021-05-14 - Candles: 1500
✅ Fetched: 2021-05-14 to 2021-06-13 - Candles: 1500
✅ Fetched: 2021-06-13 to 2021-07-13 - Candles: 1650
✅ Fetched: 2021-07-13 to 2021-08-12 - Candles: 1575
✅ Fetched: 2021-08-12 to 2021-09-11 - Candles: 1425
✅ Fetched: 2021-09-11 to 2021-10-11 - Candles: 1575
✅ Fetched: 2021-10-11 to 2021-11-10 - Candles: 1438
✅ Fetched: 2021-11-10 to 2021-12-10 - Candles: 1575
✅ Fetched: 2021-12-10 to 2022-01-09 - Candles: 1500
✅ Fetched: 2022-01-09 to 2022-02-08 - Candles: 1575
✅ Fetched: 2022-02-08 to 2022-03-10 - Candles: 1575
✅ Fetched: 2022-03-10 to 2022-04-09 - Candles: 1500
✅ Fetched: 2022-04-09 to 2022-05-09 - Candles: 1350
✅ Fetched: 2022-05-09 to 2022-06-08 - Candles: 1650
✅ Fetched: 2022-06-08 to 2022-07-08 - Candles: 1650
✅ Fetched: 2022-07-08 to 2022-08-07 - Candles: 1500
✅ Fetched: 2022-08-07 to 2022-09-06 - Candles: 1425
✅ Fetched: 2022-09-06 to 2022-10-06 - Candles: 1575
✅ Fetched: 2022-10-06 to 2022-11-05 - Candles: 1437
✅ Fetched: 2022-11-05 to 2022-12-05 - Candles: 1500
✅ Fetched: 2022-12-05 to 2023-01-04 - Candles: 1650
✅ Fetched: 2023-01-04 to 2023-02-03 - Candles: 1575
✅ Fetched: 2023-02-03 to 2023-03-05 - Candles: 1500
✅ Fetched: 2023-03-05 to 2023-04-04 - Candles: 1425
✅ Fetched: 2023-04-04 to 2023-05-04 - Candles: 1425
✅ Fetched: 2023-05-04 to 2023-06-03 - Candles: 1575
✅ Fetched: 2023-06-03 to 2023-07-03 - Candles: 1500
✅ Fetched: 2023-07-03 to 2023-08-02 - Candles: 1650
✅ Fetched: 2023-08-02 to 2023-09-01 - Candles: 1575
✅ Fetched: 2023-09-01 to 2023-10-01 - Candles: 1425
✅ Fetched: 2023-10-01 to 2023-10-31 - Candles: 1500
✅ Fetched: 2023-10-31 to 2023-11-30 - Candles: 1512
✅ Fetched: 2023-11-30 to 2023-12-30 - Candles: 1500
✅ Fetched: 2023-12-30 to 2024-01-29 - Candles: 1500
✅ Fetched: 2024-01-29 to 2024-02-28 - Candles: 1650
✅ Fetched: 2024-02-28 to 2024-03-29 - Candles: 1450
✅ Fetched: 2024-03-29 to 2024-04-28 - Candles: 1350
✅ Fetched: 2024-04-28 to 2024-05-28 - Candles: 1521
✅ Fetched: 2024-05-28 to 2024-06-27 - Candles: 1575
✅ Fetched: 2024-06-27 to 2024-07-27 - Candles: 1499
✅ Fetched: 2024-07-27 to 2024-08-26 - Candles: 1500
✅ Fetched: 2024-08-26 to 2024-09-25 - Candles: 1650
✅ Fetched: 2024-09-25 to 2024-10-25 - Candles: 1575
✅ Fetched: 2024-10-25 to 2024-11-24 - Candles: 1287
```

```
✅ Fetched: 2024-11-24 to 2024-12-24 - Candles: 1650
✅ Fetched: 2024-12-24 to 2025-01-23 - Candles: 1575
✅ Fetched: 2025-01-23 to 2025-02-22 - Candles: 1650
✅ Fetched: 2025-02-22 to 2025-03-24 - Candles: 1425
✅ Fetched: 2025-03-24 to 2025-04-18 - Candles: 1125
```

In [52]:
```
df['timestamp'] = pd.to_datetime(df['timestamp'])
df.set_index('timestamp', inplace=True)
df.index = df.index.tz_localize(None)
df.head(-1)
```

Out[52]:

timestamp	open	high	low	close	volume
2020-04-20 09:15:00	9390.20	9390.85	9269.55	9291.95	0.0
2020-04-20 09:20:00	9293.90	9322.30	9265.85	9319.20	0.0
2020-04-20 09:25:00	9319.45	9352.90	9313.55	9331.80	0.0
2020-04-20 09:30:00	9330.05	9339.95	9289.45	9296.75	0.0
2020-04-20 09:35:00	9296.00	9296.00	9244.00	9247.75	0.0
...	...	...	...	...	...
2025-04-17 15:00:00	23857.15	23872.35	23843.85	23855.80	0.0
2025-04-17 15:05:00	23855.25	23859.55	23846.65	23853.15	0.0
2025-04-17 15:10:00	23852.50	23855.75	23844.90	23847.00	0.0
2025-04-17 15:15:00	23847.20	23852.40	23841.70	23850.15	0.0
2025-04-17 15:20:00	23848.95	23852.85	23843.75	23851.65	0.0

92758 rows × 5 columns

In [53]:
```
import pandas_ta as ta

def add_all_indicators(df):
    # Trend Indicators
    df['ema_20'] = ta.ema(df['close'], length=20)
    df['ema_50'] = ta.ema(df['close'], length=50)
    df['ema_200'] = ta.ema(df['close'], length=200)
    df['sma_20'] = ta.sma(df['close'], length=20)
    df['sma_50'] = ta.sma(df['close'], length=50)

    # Momentum Indicators
    df['rsi_14'] = ta.rsi(df['close'], length=14)
    df['macd'] = ta.macd(df['close'])['MACD_12_26_9']
    df['macd_signal'] = ta.macd(df['close'])['MACDs_12_26_9']
    df['stoch_k'] = ta.stoch(df['high'], df['low'], df['close'])['STOCHk_14_3_
    df['stoch_d'] = ta.stoch(df['high'], df['low'], df['close'])['STOCHd_14_3_
    df['cci'] = ta.cci(df['high'], df['low'], df['close'], length=20)

    # Volatility Indicators
```

```
    bb = ta.bbands(df['close'], length=20)
    df['bb_upper'] = bb['BBU_20_2.0']
    df['bb_middle'] = bb['BBM_20_2.0']
    df['bb_lower'] = bb['BBL_20_2.0']
    df['atr'] = ta.atr(df['high'], df['low'], df['close'], length=14)

    # Volume-Based Indicators

    # Trend Strength
    df['adx'] = ta.adx(df['high'], df['low'], df['close'], length=14)['ADX_14'

    return df
```

In [54]:
```
df = add_all_indicators(df)
```

In [55]:
```
df.head(-1)
```

Out[55]:

timestamp	open	high	low	close	volume	ema_20	ema_
2020-04-20 09:15:00	9390.20	9390.85	9269.55	9291.95	0.0	NaN	N
2020-04-20 09:20:00	9293.90	9322.30	9265.85	9319.20	0.0	NaN	N
2020-04-20 09:25:00	9319.45	9352.90	9313.55	9331.80	0.0	NaN	N
2020-04-20 09:30:00	9330.05	9339.95	9289.45	9296.75	0.0	NaN	N
2020-04-20 09:35:00	9296.00	9296.00	9244.00	9247.75	0.0	NaN	N
...	...	...	...	...	...	...	
2025-04-17 15:00:00	23857.15	23872.35	23843.85	23855.80	0.0	23825.285378	23722.2768
2025-04-17 15:05:00	23855.25	23859.55	23846.65	23853.15	0.0	23827.939152	23727.4091
2025-04-17 15:10:00	23852.50	23855.75	23844.90	23847.00	0.0	23829.754471	23732.0989
2025-04-17 15:15:00	23847.20	23852.40	23841.70	23850.15	0.0	23831.696902	23736.7284
2025-04-17 15:20:00	23848.95	23852.85	23843.75	23851.65	0.0	23833.597197	23741.2351

92758 rows × 21 columns

In [56]:
```
df.dropna(inplace=True)
```

In [57]:
```
df.head(-1)
```

Out[57]:

timestamp	open	high	low	close	volume	ema_20	ema_
2020-04-22 13:20:00	9138.95	9162.65	9135.35	9157.10	0.0	9123.054463	9083.016
2020-04-22 13:25:00	9157.20	9159.40	9124.70	9125.70	0.0	9123.306419	9084.689
2020-04-22 13:30:00	9125.80	9126.35	9109.60	9118.00	0.0	9122.801046	9085.996
2020-04-22 13:35:00	9118.70	9130.55	9099.50	9107.75	0.0	9121.367613	9086.849
2020-04-22 13:40:00	9107.60	9117.00	9089.55	9108.95	0.0	9120.184983	9087.716
...	...	...	...	...	...	...	
2025-04-17 15:00:00	23857.15	23872.35	23843.85	23855.80	0.0	23825.285378	23722.276
2025-04-17 15:05:00	23855.25	23859.55	23846.65	23853.15	0.0	23827.939152	23727.409
2025-04-17 15:10:00	23852.50	23855.75	23844.90	23847.00	0.0	23829.754471	23732.098
2025-04-17 15:15:00	23847.20	23852.40	23841.70	23850.15	0.0	23831.696902	23736.728
2025-04-17 15:20:00	23848.95	23852.85	23843.75	23851.65	0.0	23833.597197	23741.235

92559 rows × 21 columns

In [58]:
```
def label_trades_first_hit(df, target_move=80, lookahead=5):
    """
    Labels each candle in the dataframe based on whether the price
    moves up or down by a target amount within a lookahead window.

    Parameters:
        df (pd.DataFrame): DataFrame with a 'close' column
        target_move (float): The number of points price must move to trigger a
        lookahead (int): Number of future candles to look ahead

    Returns:
```

```
        df (pd.DataFrame): Same dataframe with an added 'label' column
                           1 = Buy CE, 2 = Buy PE, 0 = No Trade
    """
    labels = []
    close_prices = df['close'].values

    for i in range(len(df) - lookahead):
        entry_price = close_prices[i]
        label = 0  # Default: No Trade

        for j in range(1, lookahead + 1):
            future_price = close_prices[i + j]

            # Check for Buy CE
            if future_price - entry_price >= target_move:
                label = 1
                break

            # Check for Buy PE
            elif entry_price - future_price >= target_move:
                label = 2
                break

        labels.append(label)

    # Pad the end with 0s (no label for incomplete future data)
    labels += [0] * lookahead
    df['label'] = labels

    return df
```

In [59]:
```
df = label_trades_first_hit(df, target_move=80, lookahead=5)
```

In [60]:
```
df.head(-1)
```

Out[60]:

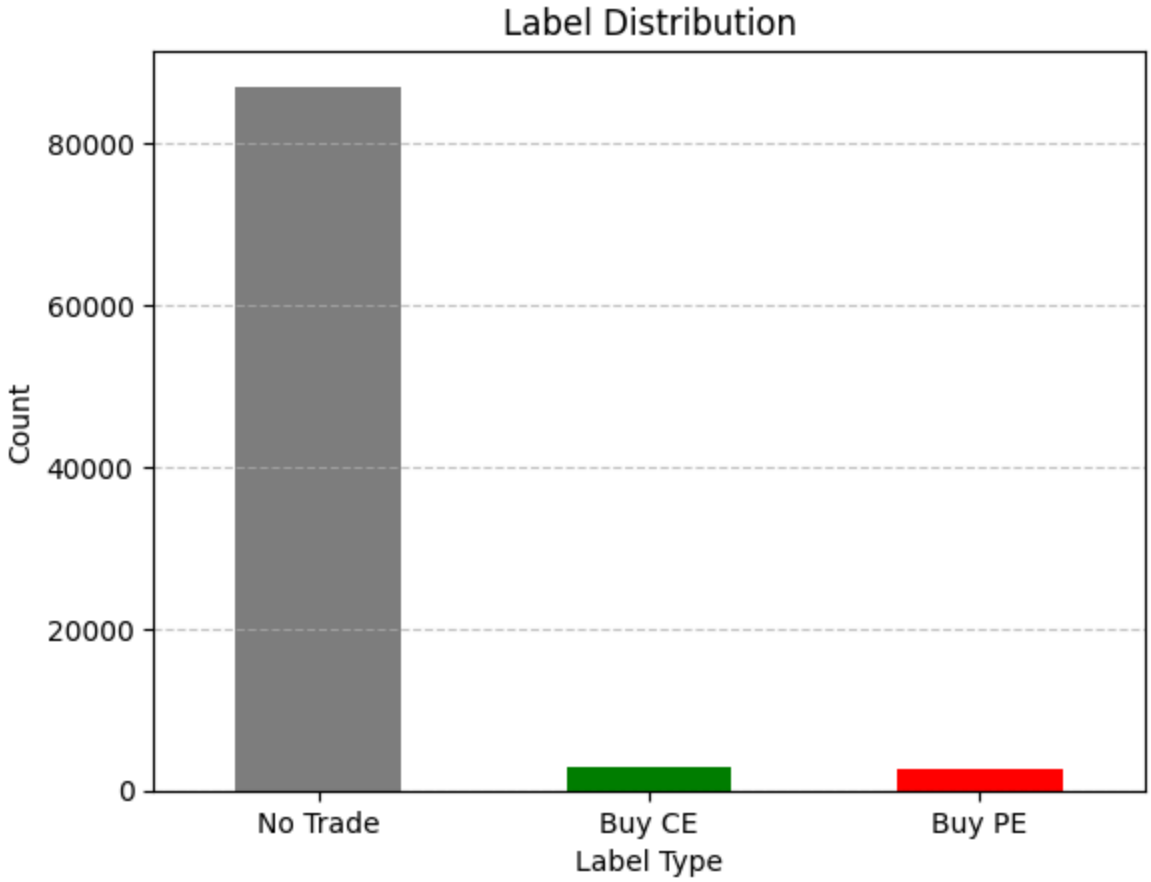
timestamp	open	high	low	close	volume	ema_20	ema_
2020-04-22 13:20:00	9138.95	9162.65	9135.35	9157.10	0.0	9123.054463	9083.016
2020-04-22 13:25:00	9157.20	9159.40	9124.70	9125.70	0.0	9123.306419	9084.689
2020-04-22 13:30:00	9125.80	9126.35	9109.60	9118.00	0.0	9122.801046	9085.996
2020-04-22 13:35:00	9118.70	9130.55	9099.50	9107.75	0.0	9121.367613	9086.849
2020-04-22 13:40:00	9107.60	9117.00	9089.55	9108.95	0.0	9120.184983	9087.716
...	...	...	...	...	...	...	
2025-04-17 15:00:00	23857.15	23872.35	23843.85	23855.80	0.0	23825.285378	23722.276
2025-04-17 15:05:00	23855.25	23859.55	23846.65	23853.15	0.0	23827.939152	23727.409
2025-04-17 15:10:00	23852.50	23855.75	23844.90	23847.00	0.0	23829.754471	23732.098
2025-04-17 15:15:00	23847.20	23852.40	23841.70	23850.15	0.0	23831.696902	23736.728
2025-04-17 15:20:00	23848.95	23852.85	23843.75	23851.65	0.0	23833.597197	23741.235

92559 rows × 22 columns

In [62]:
```
label_counts = df['label'].value_counts().sort_index()

# Print nicely
label_map = {0: 'No Trade', 1: 'Buy CE', 2: 'Buy PE'}
for label, count in label_counts.items():
    print(f"{label_map[label]} (Label {label}): {count}")

import matplotlib.pyplot as plt

label_counts.rename(index=label_map).plot(kind='bar', color=['gray', 'green',
plt.title('Label Distribution')
```

```
plt.ylabel('Count')
plt.xlabel('Label Type')
plt.xticks(rotation=0)
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.show()
```

```
No Trade (Label 0): 87086
Buy CE (Label 1): 2825
Buy PE (Label 2): 2649
```



🔁 Phase-by-Phase LSTM Workflow in Jupyter

In [69]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import tensorflow as tf
from sklearn.model_selection import train_test_split
from sklearn.utils.class_weight import compute_class_weight
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import classification_report, confusion_matrix
from imblearn.over_sampling import BorderlineSMOTE
from tensorflow.keras.models import Model
from tensorflow.keras.layers import (Input, LSTM, Dense, Dropout, BatchNormali
                                     Bidirectional, Lambda, Flatten, Activatic
                                     RepeatVector, Permute, Multiply)
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau, Model
from tensorflow.keras.optimizers import Adam
import tensorflow.keras.backend as K
```

In [70]:
```
features = [
    'close', 'open', 'high', 'low',
    'ema_200', 'rsi_14', 'macd', 'adx',
    'bb_upper', 'bb_lower', 'stoch_k', 'stoch_d'
]

scaler = MinMaxScaler()
df[features] = scaler.fit_transform(df[features])
```

In [71]:
```
SEQ_LEN = 60
X, y = [], []

for i in range(SEQ_LEN, len(df)):
    X.append(df[features].iloc[i-SEQ_LEN:i].values)
    y.append(df['label'].iloc[i])  # 🔄 using 'label' now

X = np.array(X)
y = np.array(y)
```

In [72]:
```
X_flat = X.reshape(X.shape[0], -1)
smote = BorderlineSMOTE(kind='borderline-2', random_state=42)
X_res, y_res = smote.fit_resample(X_flat, y)
X_res = X_res.reshape(-1, SEQ_LEN, len(features))
```

```
/Users/nil/Library/Python/3.9/lib/python/site-packages/sklearn/base.py:474: Fu
tureWarning: `BaseEstimator._validate_data` is deprecated in 1.6 and will be r
emoved in 1.7. Use `sklearn.utils.validation.validate_data` instead. This func
tion becomes public and is part of the scikit-learn developer API.
  warnings.warn(
```

In [78]:
```
# ✅ Attention Block (unchanged)
def attention_block_with_weights(inputs):
    attention_scores = Dense(1, activation='tanh')(inputs)
    attention_scores = Flatten()(attention_scores)
    attention_weights = Activation('softmax', name='attention_weights')(attent
    attention_weights = RepeatVector(inputs.shape[-1])(attention_weights)
    attention_weights = Permute([2, 1])(attention_weights)
```

```
    context_vector = Multiply()([inputs, attention_weights])
    return context_vector, attention_weights
```

In [79]:
```
input_shape = (X_res.shape[1], X_res.shape[2])
inp = Input(shape=input_shape)

x = Bidirectional(LSTM(128, return_sequences=True))(inp)
x = BatchNormalization()(x)
x = Dropout(0.3)(x)

x = Bidirectional(LSTM(64, return_sequences=True))(x)
x = BatchNormalization()(x)
x = Dropout(0.3)(x)

# ✅ Keep attention for visualization (but don't output it)
x, att_weights = attention_block_with_weights(x)
x = Lambda(lambda x: K.sum(x, axis=1))(x)

x = Dense(64, activation='relu')(x)
x = Dropout(0.3)(x)
output = Dense(3, activation='softmax')(x)

# ✅ This model only outputs class predictions
model = Model(inputs=inp, outputs=output)

# Compile
model.compile(optimizer=Adam(0.0005), loss='sparse_categorical_crossentropy',

# Print summary
model.summary()
```

Model: "functional_1"

Layer (type)	Output Shape	Param #	Connected to
input_layer_1 (InputLayer)	(None, 60, 12)	0	–
bidirectional_2 (Bidirectional)	(None, 60, 256)	144,384	input_layer_1[0
batch_normalizatio… (BatchNormalizatio…	(None, 60, 256)	1,024	bidirectional_2
dropout_3 (Dropout)	(None, 60, 256)	0	batch_normaliza
bidirectional_3 (Bidirectional)	(None, 60, 128)	164,352	dropout_3[0][0]
batch_normalizatio… (BatchNormalizatio…	(None, 60, 128)	512	bidirectional_3
dropout_4 (Dropout)	(None, 60, 128)	0	batch_normaliza
dense_2 (Dense)	(None, 60, 1)	129	dropout_4[0][0]
flatten_1 (Flatten)	(None, 60)	0	dense_2[0][0]
attention_weights (Activation)	(None, 60)	0	flatten_1[0][0]
repeat_vector_1 (RepeatVector)	(None, 128, 60)	0	attention_weigh
permute_1 (Permute)	(None, 60, 128)	0	repeat_vector_1
multiply_1 (Multiply)	(None, 60, 128)	0	dropout_4[0][0] permute_1[0][0]
lambda_1 (Lambda)	(None, 128)	0	multiply_1[0][0
dense_3 (Dense)	(None, 64)	8,256	lambda_1[0][0]
dropout_5 (Dropout)	(None, 64)	0	dense_3[0][0]
dense_4 (Dense)	(None, 3)	195	dropout_5[0][0]

Total params: 318,852 (1.22 MB)

Trainable params: 318,084 (1.21 MB)

Non-trainable params: 768 (3.00 KB)

In [80]:
```
X_train, X_val, y_train, y_val = train_test_split(X_res, y_res, test_size=0.2,
class_weights = compute_class_weight(class_weight='balanced', classes=np.uniqu
class_weights = dict(enumerate(class_weights))
```

In [81]:
```
early_stop = EarlyStopping(monitor='val_loss', patience=5, restore_best_weight
reduce_lr = ReduceLROnPlateau(monitor='val_loss', factor=0.5, patience=3, verb
checkpoint = ModelCheckpoint("best_lstm_attention_model.keras", monitor='val_l
```

```
history = model.fit(
    X_train, y_train,
    validation_data=(X_val, y_val),
    epochs=60,
    batch_size=64,
    class_weight=class_weights,
    callbacks=[early_stop, reduce_lr, checkpoint],
    verbose=1
)
```

```
Epoch 1/60
```

```
2025-04-18 21:37:34.282159: I tensorflow/core/grappler/optimizers/custom_graph
_optimizer_registry.cc:117] Plugin optimizer for device_type GPU is enabled.
```

```
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 145ms/step - accuracy: 0.4247 - loss: 1.1026
Epoch 1: val_loss improved from inf to 0.88833, saving model to best_lstm_atte
ntion_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 526s 160ms/step - accuracy: 0.4247 - loss: 1.10
26 - val_accuracy: 0.5865 - val_loss: 0.8883 - learning_rate: 5.0000e-04
Epoch 2/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 137ms/step - accuracy: 0.6366 - loss: 0.7951
Epoch 2: val_loss improved from 0.88833 to 0.64937, saving model to best_lstm_
attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 496s 152ms/step - accuracy: 0.6366 - loss: 0.79
51 - val_accuracy: 0.7117 - val_loss: 0.6494 - learning_rate: 5.0000e-04
Epoch 3/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 139ms/step - accuracy: 0.7515 - loss: 0.5944
Epoch 3: val_loss improved from 0.64937 to 0.46329, saving model to best_lstm_
attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 503s 154ms/step - accuracy: 0.7515 - loss: 0.59
44 - val_accuracy: 0.8172 - val_loss: 0.4633 - learning_rate: 5.0000e-04
Epoch 4/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 138ms/step - accuracy: 0.8013 - loss: 0.5002
Epoch 4: val_loss improved from 0.46329 to 0.39155, saving model to best_lstm_
attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 503s 154ms/step - accuracy: 0.8013 - loss: 0.50
02 - val_accuracy: 0.8576 - val_loss: 0.3916 - learning_rate: 5.0000e-04
Epoch 5/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 150ms/step - accuracy: 0.8262 - loss: 0.4508
Epoch 5: val_loss did not improve from 0.39155
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 540s 165ms/step - accuracy: 0.8262 - loss: 0.45
08 - val_accuracy: 0.8485 - val_loss: 0.4096 - learning_rate: 5.0000e-04
Epoch 6/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 139ms/step - accuracy: 0.8473 - loss: 0.4080
Epoch 6: val_loss improved from 0.39155 to 0.33489, saving model to best_lstm_
attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 504s 154ms/step - accuracy: 0.8473 - loss: 0.40
80 - val_accuracy: 0.8779 - val_loss: 0.3349 - learning_rate: 5.0000e-04
Epoch 7/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 138ms/step - accuracy: 0.8608 - loss: 0.3797
Epoch 7: val_loss improved from 0.33489 to 0.31911, saving model to best_lstm_
attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 503s 154ms/step - accuracy: 0.8608 - loss: 0.37
97 - val_accuracy: 0.8881 - val_loss: 0.3191 - learning_rate: 5.0000e-04
Epoch 8/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 146ms/step - accuracy: 0.8741 - loss: 0.3540
Epoch 8: val_loss improved from 0.31911 to 0.26861, saving model to best_lstm_
attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 526s 161ms/step - accuracy: 0.8741 - loss: 0.35
40 - val_accuracy: 0.9035 - val_loss: 0.2686 - learning_rate: 5.0000e-04
Epoch 9/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 144ms/step - accuracy: 0.8662 - loss: 0.3710
Epoch 9: val_loss did not improve from 0.26861
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 518s 159ms/step - accuracy: 0.8662 - loss: 0.37
10 - val_accuracy: 0.8992 - val_loss: 0.2806 - learning_rate: 5.0000e-04
Epoch 10/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 144ms/step - accuracy: 0.8846 - loss: 0.3339
Epoch 10: val_loss did not improve from 0.26861
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 520s 159ms/step - accuracy: 0.8846 - loss: 0.33
39 - val_accuracy: 0.9037 - val_loss: 0.2885 - learning_rate: 5.0000e-04
```

```
Epoch 11/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 145ms/step - accuracy: 0.8926 - loss: 0.3150
Epoch 11: val_loss improved from 0.26861 to 0.24734, saving model to best_lstm
_attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 523s 160ms/step - accuracy: 0.8926 - loss: 0.31
50 - val_accuracy: 0.9248 - val_loss: 0.2473 - learning_rate: 5.0000e-04
Epoch 12/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 140ms/step - accuracy: 0.8953 - loss: 0.3142
Epoch 12: val_loss did not improve from 0.24734
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 504s 155ms/step - accuracy: 0.8953 - loss: 0.31
42 - val_accuracy: 0.9173 - val_loss: 0.2519 - learning_rate: 5.0000e-04
Epoch 13/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 145ms/step - accuracy: 0.8988 - loss: 0.3087
Epoch 13: val_loss did not improve from 0.24734
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 523s 160ms/step - accuracy: 0.8988 - loss: 0.30
87 - val_accuracy: 0.9235 - val_loss: 0.2584 - learning_rate: 5.0000e-04
Epoch 14/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 142ms/step - accuracy: 0.9038 - loss: 0.2992
Epoch 14: ReduceLROnPlateau reducing learning rate to 0.0002500000118743628.

Epoch 14: val_loss did not improve from 0.24734
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 514s 158ms/step - accuracy: 0.9038 - loss: 0.29
92 - val_accuracy: 0.9019 - val_loss: 0.3254 - learning_rate: 5.0000e-04
Epoch 15/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 138ms/step - accuracy: 0.9325 - loss: 0.2206
Epoch 15: val_loss improved from 0.24734 to 0.18841, saving model to best_lstm
_attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 500s 153ms/step - accuracy: 0.9325 - loss: 0.22
06 - val_accuracy: 0.9439 - val_loss: 0.1884 - learning_rate: 2.5000e-04
Epoch 16/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 142ms/step - accuracy: 0.9357 - loss: 0.2084
Epoch 16: val_loss improved from 0.18841 to 0.16207, saving model to best_lstm
_attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 514s 157ms/step - accuracy: 0.9357 - loss: 0.20
84 - val_accuracy: 0.9560 - val_loss: 0.1621 - learning_rate: 2.5000e-04
Epoch 17/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 137ms/step - accuracy: 0.9402 - loss: 0.1999
Epoch 17: val_loss did not improve from 0.16207
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 498s 153ms/step - accuracy: 0.9402 - loss: 0.19
99 - val_accuracy: 0.9543 - val_loss: 0.1656 - learning_rate: 2.5000e-04
Epoch 18/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 139ms/step - accuracy: 0.9390 - loss: 0.2048
Epoch 18: val_loss improved from 0.16207 to 0.16089, saving model to best_lstm
_attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 504s 154ms/step - accuracy: 0.9390 - loss: 0.20
48 - val_accuracy: 0.9578 - val_loss: 0.1609 - learning_rate: 2.5000e-04
Epoch 19/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 150ms/step - accuracy: 0.9415 - loss: 0.1954
Epoch 19: val_loss improved from 0.16089 to 0.16040, saving model to best_lstm
_attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 540s 166ms/step - accuracy: 0.9415 - loss: 0.19
54 - val_accuracy: 0.9581 - val_loss: 0.1604 - learning_rate: 2.5000e-04
Epoch 20/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 149ms/step - accuracy: 0.9442 - loss: 0.1910
Epoch 20: val_loss improved from 0.16040 to 0.15119, saving model to best_lstm
_attention_model.keras
```

```
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 537s 164ms/step - accuracy: 0.9442 - loss: 0.19
10 - val_accuracy: 0.9608 - val_loss: 0.1512 - learning_rate: 2.5000e-04
Epoch 21/60
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 0s 146ms/step - accuracy: 0.9449 - loss: 0.1872
Epoch 21: val_loss improved from 0.15119 to 0.14453, saving model to best_lstm
_attention_model.keras
3264/3264 ━━━━━━━━━━━━━━━━━━━━ 525s 161ms/step - accuracy: 0.9449 - loss: 0.18
72 - val_accuracy: 0.9629 - val_loss: 0.1445 - learning_rate: 2.5000e-04
Epoch 22/60
 784/3264 ━━━━━━━━━━━━━━━━━━━━ 6:05 148ms/step - accuracy: 0.9477 - loss: 0.18
01
```

```
---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
Cell In[81], line 5
      2 reduce_lr = ReduceLROnPlateau(monitor='val_loss', factor=0.5, patience
=3, verbose=1)
      3 checkpoint = ModelCheckpoint("best_lstm_attention_model.keras", monito
r='val_loss', save_best_only=True, verbose=1)
----> 5 history = model.fit(
      6     X_train, y_train,
      7     validation_data=(X_val, y_val),
      8     epochs=60,
      9     batch_size=64,
     10     class_weight=class_weights,
     11     callbacks=[early_stop, reduce_lr, checkpoint],
     12     verbose=1
     13 )

File ~/Library/Python/3.9/lib/python/site-packages/keras/src/utils/traceback_u
tils.py:117, in filter_traceback.<locals>.error_handler(*args, **kwargs)
    115 filtered_tb = None
    116 try:
--> 117     return fn(*args, **kwargs)
    118 except Exception as e:
    119     filtered_tb = _process_traceback_frames(e.__traceback__)

File ~/Library/Python/3.9/lib/python/site-packages/keras/src/backend/tensorflo
w/trainer.py:368, in TensorFlowTrainer.fit(self, x, y, batch_size, epochs, ver
bose, callbacks, validation_split, validation_data, shuffle, class_weight, sam
ple_weight, initial_epoch, steps_per_epoch, validation_steps, validation_batch
_size, validation_freq)
    366 for step, iterator in epoch_iterator:
    367     callbacks.on_train_batch_begin(step)
--> 368     logs = self.train_function(iterator)
    369     callbacks.on_train_batch_end(step, logs)
    370     if self.stop_training:

File ~/Library/Python/3.9/lib/python/site-packages/keras/src/backend/tensorflo
w/trainer.py:216, in TensorFlowTrainer._make_function.<locals>.function(iterat
or)
    212 def function(iterator):
    213     if isinstance(
    214         iterator, (tf.data.Iterator, tf.distribute.DistributedIterato
r)
    215     ):
--> 216         opt_outputs = multi_step_on_iterator(iterator)
    217         if not opt_outputs.has_value():
    218             raise StopIteration

File ~/Library/Python/3.9/lib/python/site-packages/tensorflow/python/util/trac
eback_utils.py:150, in filter_traceback.<locals>.error_handler(*args, **kwarg
s)
    148 filtered_tb = None
    149 try:
--> 150   return fn(*args, **kwargs)
    151 except Exception as e:
    152   filtered_tb = _process_traceback_frames(e.__traceback__)
```

```
File ~/Library/Python/3.9/lib/python/site-packages/tensorflow/python/eager/polymorphic_function/polymorphic_function.py:833, in Function.__call__(self, *args, **kwds)
    830 compiler = "xla" if self._jit_compile else "nonXla"
    832 with OptionalXlaContext(self._jit_compile):
--> 833   result = self._call(*args, **kwds)
    835 new_tracing_count = self.experimental_get_tracing_count()
    836 without_tracing = (tracing_count == new_tracing_count)

File ~/Library/Python/3.9/lib/python/site-packages/tensorflow/python/eager/polymorphic_function/polymorphic_function.py:878, in Function._call(self, *args, **kwds)
    875 self._lock.release()
    876 # In this case we have not created variables on the first call. So we can
    877 # run the first trace but we should fail if variables are created.
--> 878 results = tracing_compilation.call_function(
    879     args, kwds, self._variable_creation_config
    880 )
    881 if self._created_variables:
    882   raise ValueError("Creating variables on a non-first call to a function"
    883                    " decorated with tf.function.")

File ~/Library/Python/3.9/lib/python/site-packages/tensorflow/python/eager/polymorphic_function/tracing_compilation.py:139, in call_function(args, kwargs, tracing_options)
    137 bound_args = function.function_type.bind(*args, **kwargs)
    138 flat_inputs = function.function_type.unpack_inputs(bound_args)
--> 139 return function._call_flat(  # pylint: disable=protected-access
    140     flat_inputs, captured_inputs=function.captured_inputs
    141 )

File ~/Library/Python/3.9/lib/python/site-packages/tensorflow/python/eager/polymorphic_function/concrete_function.py:1322, in ConcreteFunction._call_flat(self, tensor_inputs, captured_inputs)
   1318 possible_gradient_type = gradients_util.PossibleTapeGradientTypes(args)
   1319 if (possible_gradient_type == gradients_util.POSSIBLE_GRADIENT_TYPES_NONE
   1320     and executing_eagerly):
   1321   # No tape is watching; skip to running the function.
-> 1322   return self._inference_function.call_preflattened(args)
   1323 forward_backward = self._select_forward_and_backward_functions(
   1324     args,
   1325     possible_gradient_type,
   1326     executing_eagerly)
   1327 forward_function, args_with_tangents = forward_backward.forward()

File ~/Library/Python/3.9/lib/python/site-packages/tensorflow/python/eager/polymorphic_function/atomic_function.py:216, in AtomicFunction.call_preflattened(self, args)
    214 def call_preflattened(self, args: Sequence[core.Tensor]) -> Any:
    215   """Calls with flattened tensor inputs and returns the structured output."""
```

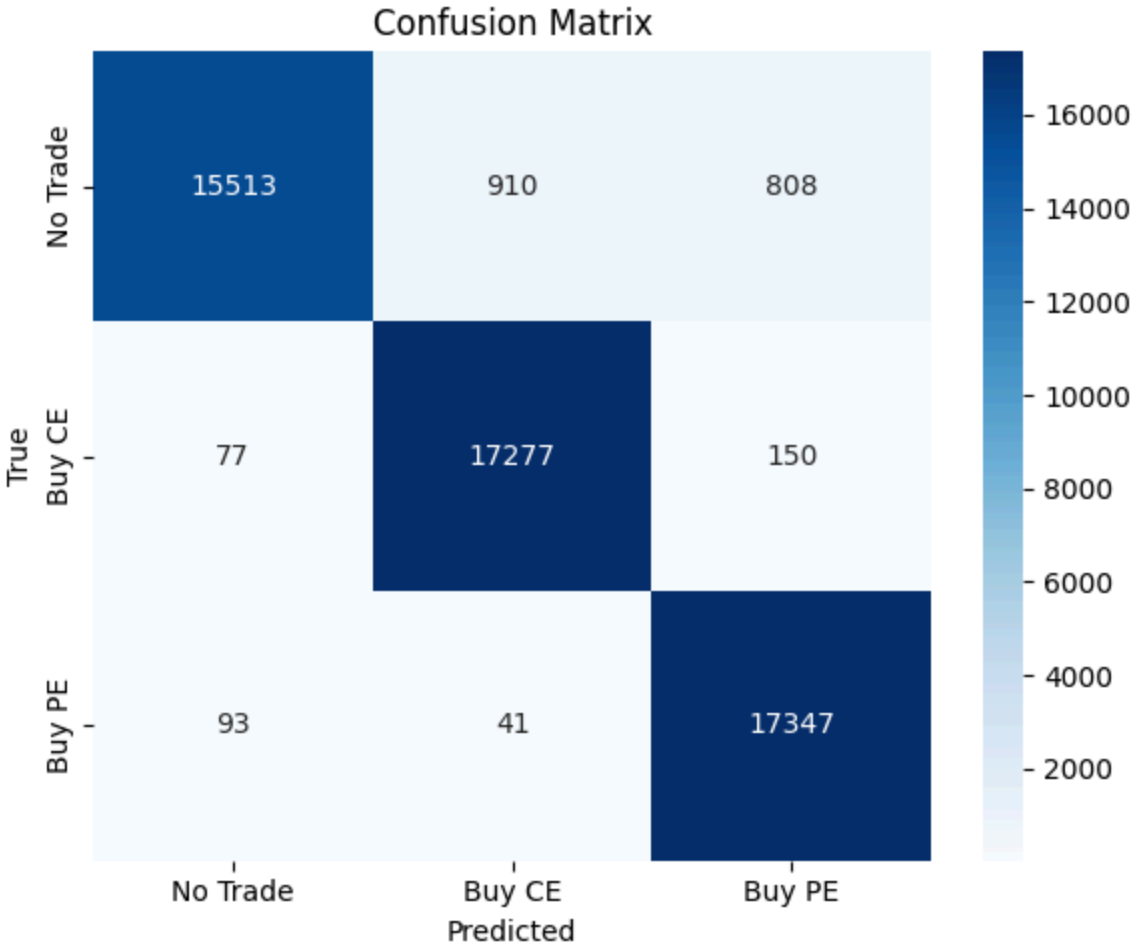
```
--> 216   flat_outputs = self.call_flat(*args)
    217   return self.function_type.pack_output(flat_outputs)

File ~/Library/Python/3.9/lib/python/site-packages/tensorflow/python/eager/polymorphic_function/atomic_function.py:251, in AtomicFunction.call_flat(self, *args)
    249 with record.stop_recording():
    250   if self._bound_context.executing_eagerly():
--> 251     outputs = self._bound_context.call_function(
    252         self.name,
    253         list(args),
    254         len(self.function_type.flat_outputs),
    255     )
    256   else:
    257     outputs = make_call_op_in_graph(
    258         self,
    259         list(args),
    260         self._bound_context.function_call_options.as_attrs(),
    261     )

File ~/Library/Python/3.9/lib/python/site-packages/tensorflow/python/eager/context.py:1500, in Context.call_function(self, name, tensor_inputs, num_outputs)
   1498 cancellation_context = cancellation.context()
   1499 if cancellation_context is None:
-> 1500   outputs = execute.execute(
   1501       name.decode("utf-8"),
   1502       num_outputs=num_outputs,
   1503       inputs=tensor_inputs,
   1504       attrs=attrs,
   1505       ctx=self,
   1506   )
   1507 else:
   1508   outputs = execute.execute_with_cancellation(
   1509       name.decode("utf-8"),
   1510       num_outputs=num_outputs,
   (...)
   1514       cancellation_manager=cancellation_context,
   1515   )

File ~/Library/Python/3.9/lib/python/site-packages/tensorflow/python/eager/execute.py:53, in quick_execute(op_name, num_outputs, inputs, attrs, ctx, name)
     51 try:
     52   ctx.ensure_initialized()
---> 53   tensors = pywrap_tfe.TFE_Py_Execute(ctx._handle, device_name, op_name,
     54                                       inputs, attrs, num_outputs)
     55 except core._NotOkStatusException as e:
     56   if name is not None:

KeyboardInterrupt:
```

In [82]:
```
att_model = Model(inputs=model.input, outputs=att_weights)
att_model.save("attention_visualizer_model.keras")
```

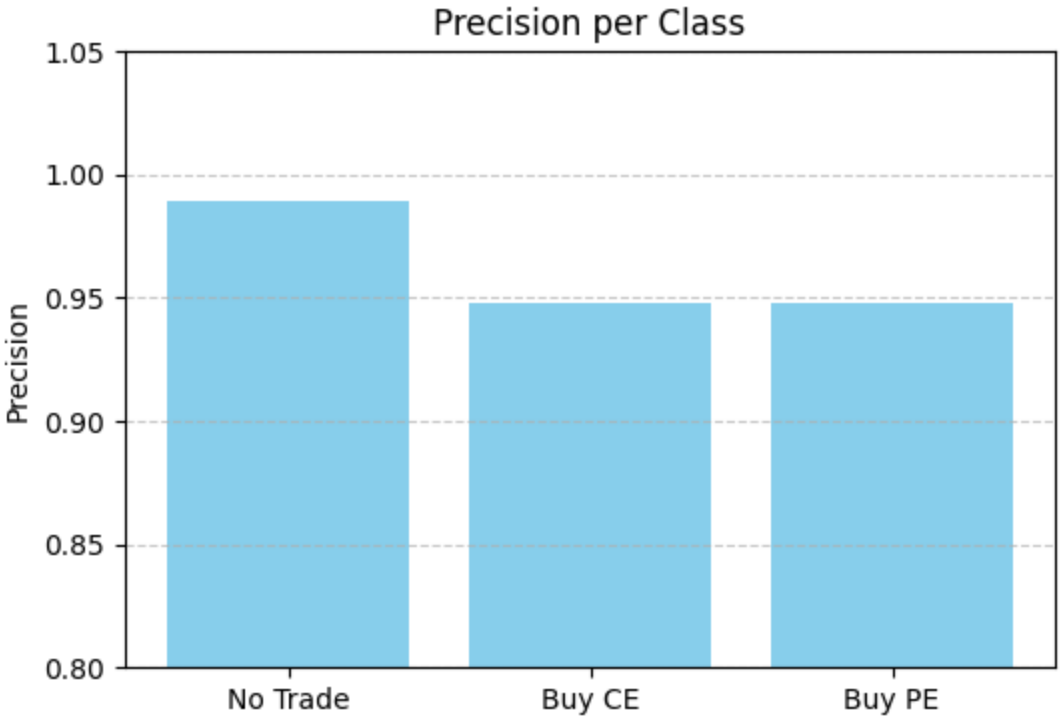
In [94]:
```
# Predict on validation set
y_pred_probs = model.predict(X_val)  # ✅ Only one output
```

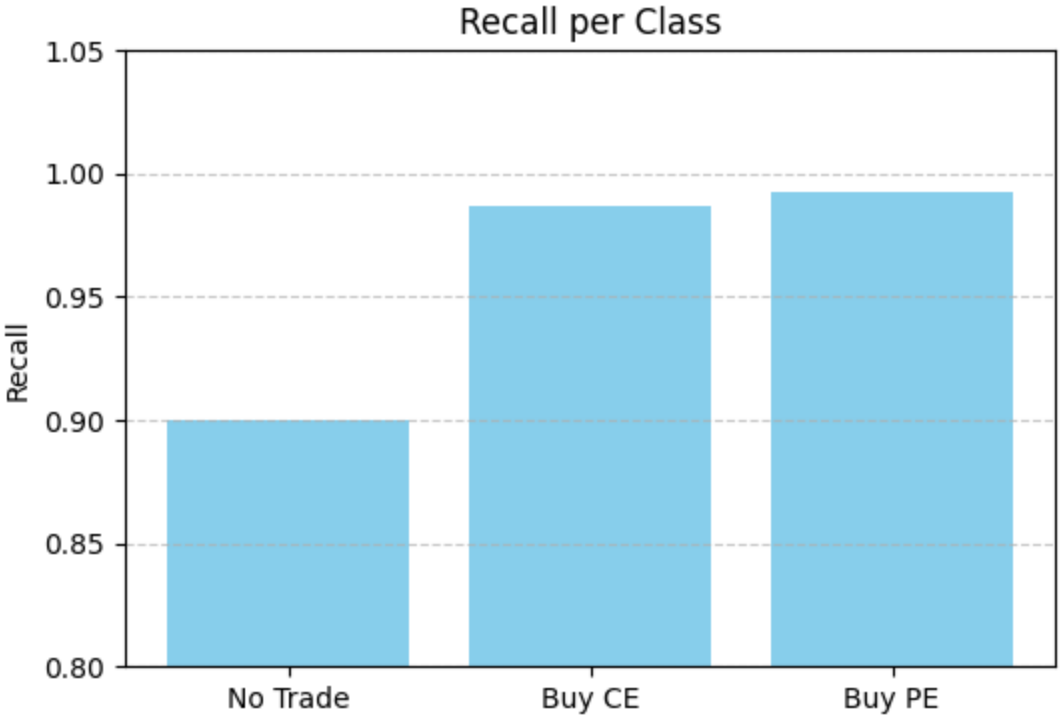
```
y_pred = np.argmax(y_pred_probs, axis=1)

# Classification Report
print("Classification Report:")
print(classification_report(y_val, y_pred, target_names=["No Trade", "Buy CE",

# Confusion Matrix
cm = confusion_matrix(y_val, y_pred)
plt.figure(figsize=(6, 5))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues',
            xticklabels=["No Trade", "Buy CE", "Buy PE"],
            yticklabels=["No Trade", "Buy CE", "Buy PE"])
plt.xlabel("Predicted")
plt.ylabel("True")
plt.title("Confusion Matrix")
plt.tight_layout()
plt.show()
```

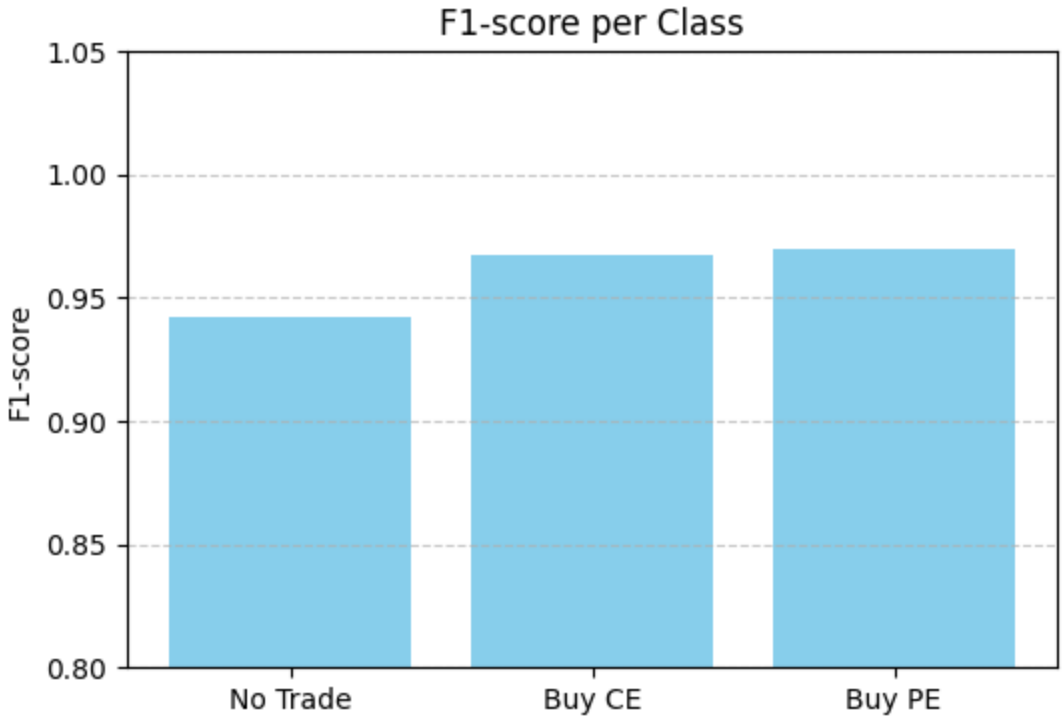
```
1632/1632 ━━━━━━━━━━━━━━━━━━━━ 58s 35ms/step
Classification Report:
              precision    recall  f1-score   support

    No Trade       0.99      0.90      0.94     17231
      Buy CE       0.95      0.99      0.97     17504
      Buy PE       0.95      0.99      0.97     17481

    accuracy                           0.96     52216
   macro avg       0.96      0.96      0.96     52216
weighted avg       0.96      0.96      0.96     52216
```



In [97]:
```
import matplotlib.pyplot as plt
from sklearn.metrics import classification_report

report = classification_report(y_val, y_pred, target_names=["No Trade", "Buy C

metrics = ["precision", "recall", "f1-score"]
classes = ["No Trade", "Buy CE", "Buy PE"]

for metric in metrics:
    plt.figure(figsize=(6,4))
    values = [report[c][metric] for c in classes]
    plt.bar(classes, values, color="skyblue")
    plt.ylim(0.8, 1.05)
    plt.title(f"{metric.capitalize()} per Class")
    plt.ylabel(metric.capitalize())
    plt.grid(True, axis='y', linestyle='--', alpha=0.7)
    plt.show()
```


Precision per Class
Precision
1.05
1.00
0.95
0.90
0.85
0.80
No Trade
Buy CE
Buy PE


Recall per Class
Recall
1.05
1.00
0.95
0.90
0.85
0.80
No Trade
Buy CE
Buy PE



In [101…

```
plt.figure(figsize=(16,4))
plt.plot(y_val[:200], label="Actual", linewidth=2)
plt.plot(y_pred[:200], label="Predicted", linestyle='--')
plt.title("Actual vs Predicted Class (First 200 Samples)")
plt.xlabel("Sample Index")
plt.ylabel("Class Label")
plt.legend()
plt.grid(True)
plt.show()
```